Modeling and Designing A PID Controlled Magnetic Levitation System

By: Stonewall Hickman

Project Advisor: Dr. David Gore

Abstract

This project was designed to tackle the concept of magnetic levitation by building a device that could stably levitate a permanent magnet within an electromagnetic field. Two magnets are not capable of producing levitation by themselves due to the fact that they cannot counteract the gravitational forces needed to achieve an equilibrium of levitation. In order to do this, a Hall effect sensor was used to detect the magnet's position with respect to the electromagnet: A PID (proportional, integral, derivative) algorithm was used to control the position of the magnet in the electromagnetic field, and a circuit was constructed to send the information to the system so that it can be controlled. The algorithm was written and designed into an Arduino microcontroller to be used to process information in the circuit. The results of this experiment showed that the magnet was able to be stably levitated at a desired set point (0.95- 1.4 cm from electromagnet) under these conditions using the proportional, derivative, and integral terms of the algorithm seen in the data section below.

Introduction

In this project, magnetic levitation is a method to suspend a magnet with only the use of magnetic fields. This idea can be seen in the Maglev trains of Japan where the trains use electromagnets to create lift and propulsion, this in turn, reduces the friction along the track of and allows the train to glide across the track. Magnetic levitation did not always seem possible r due to Samuel Earnshaw's Theorem of 1842. His theorem states that it is impossible to levitate permanent magnets (ferromagnets) because of the fact that given two repelling magnets, one would eventually flip over due to the torque they exert on each other and thus could not achieve a stable levitation. Earnshaw's Theorem can be circumvented via servomechanisms, which measure the speed and position of an object, and use a feedback loop to control error. This project is designed to implement this result by using a PID algorithm, a circuit, a Hall effect sensor, and the arduino microcontroller that it will be possible to levitate a magnet in an electromagnetic field. The choice to build and test the project this way was motivated by its applications in computer science, physics, and engineering.

Theory

Analytical model

In order to understand the physical system, it must be broken up into; the electrical and mechanical parts.. The following electrical part of the system is shown below:

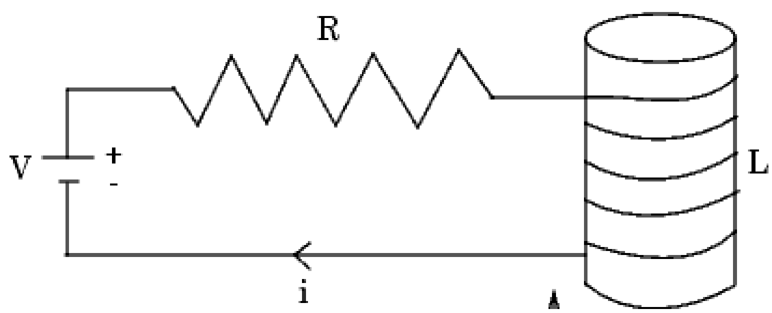

Figure 1.1: Electrical description of the magnetic levitation system

The electromagnet requires an input voltage V to produce a magnetic field. Kirchhoff's voltage law is used to find a relationship between the physical quantities:

$$V = V_R + V_L \tag{1.1}$$

Where V_R is the voltage across the resistor and V_L is the voltage across the inductor(electromagnet). V_R and V_L can be broken further to show that,

$$V = iR_s + L\frac{dI}{dt} \tag{1.2}$$

Where R_s is the resistance in series with the electromagnet, L is the inductance of the electromagnet, $\frac{di}{dt}$ is the rate of change of the current with respect to time.

The mechanical model illustrates the interaction of the forces in magnetic levitation. The following model is shown in figure 1.2 below:

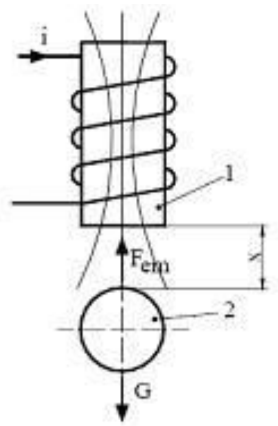

Figure 1.2: Mechanical description of the magnetic levitation system

The electromagnet receives a current that varies as the position varies so that it can levitate the magnet in the electromagnetic field. Using Newton's second law, the sum of the forces between the two magnets are represented as:

$$F = F_g - F_{em} \tag{1.3}$$

Replacing these equations with the actual quantities, this can be rewritten as the following:

$$m\frac{d^2x}{dt^2} = mg - C\left(\frac{i^2}{x^2}\right) \tag{1.4}$$

The result is a second order nonlinear differential equation that depends on the current, *i*, & position *x* of the electromagnet and a magnetic force constant C that is experimentally determined. Equation 1.4 can be determined and proven by looking at energy properties of inductors and geometry of the system. These equations will become important in modeling the system.

State Space Modeling

The state space model stems from control theory and has many applications in engineering. It is important for this project because it can be used to model the physical equations of the magnetic levitation system. A state-space model represents a system by a series of first-order differential state equations and algebraic output equations. Linear systems of the state space model can be written as the following

$$\dot{\vec{x}} = A\vec{x} + B\vec{u} \quad (1.5)$$

$$\vec{y} = C\vec{x}$$

Where $\dot{\vec{x}}$ represents time invariant vector matrix of the system, otherwise known as the state vector. The matrix A represents a matrix of state variables (current state of the system $\vec{x}$), B is the input matrix of input states ($\vec{u}$), and C is the output matrix. Notice that equation 1.4 is a nonlinear system and can be converted into a linearized system using linearization techniques. To get the equations in the form of a state space model, the system can be written as

$$m\frac{dv}{dt} = mg - C\left(\frac{i^2}{x^2}\right)$$

$$u = iR_s + L\frac{dI}{dt}$$

Where now the system is represented as a system of first order differential equations. By setting

$$x_1 = x(t), x_2 = \dot{x}(t), x_3 = i(t)$$

the system can be written to match the state space model

$$\dot{x}_1 = x_2(t)$$

$$\dot{x}_2 = g - \frac{C}{m}\left(\frac{x_3}{x_1}\right)^2$$

$$\dot{x}_3 = \frac{1}{L}(u - Rx_3)$$

Using a Taylor series approximation around the equilibrium point of the system, the state space model is shown as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{2Cx_3{}^2}{mx_1{}^3} & 0 & -2\frac{Cx_3{}^2}{mx_1{}^3} \\ 0 & 0 & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} u \qquad (1.6)$$

$$[y] = [1 \quad 0 \quad 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

The state space model is used to plot the data of the system in Matlab. For the purpose of this experiment, it will only be used to show the effect of a PID controller on the system, but has many more applications outside of this experiment.

Position detection

In order to determine the location of the magnet with respect to the electromagnet, a Hall effect sensor will be used to measure the relative position. A Hall effect sensor outputs a voltage called the Hall voltage when a magnet is brought near the sensor.

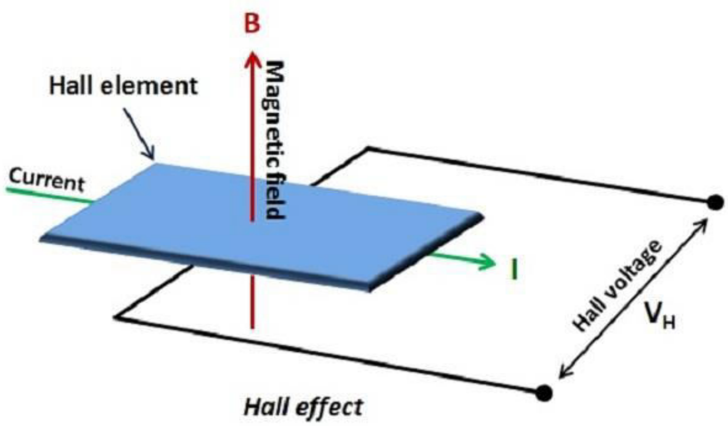

Figure 1.3: General picture of a hall effect sensor

A voltmeter can measure the Hall voltage when placed across the sensor. The Hall voltage can also be approximated by the following equation:

$$V_H = R_H \left(\frac{IB}{t}\right) \tag{1.7}$$

V_H is the hall voltage, I is the current flowing through the sensor, B is the magnetic field, t is the sensor thickness, and R_H is the Hall effect coefficient. The sensor value is important because it is used to as an input to the PID control algorithm to control position

Control algorithm

Because of the nonlinear nature of the magnetic levitation system and the existance of Earnshaw's theorem, a control algorithm is needed to control the magnet's strength. The control algorithm used in this experiment is a called a Proportional, Integral, Derivative (PID) controller which calculates an error value between a set point (SP) and current position (CP) in a feedback loop.

$$e(t) = SP - CP \qquad (1.8)$$

The algorithm adjusts the strength of the electromagnet to minimize the error so that the magnet will levitate at the desired set point. The PID algorithm is given as

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \qquad (1.9)$$

Where u(t) is the controller output and $K_p, K_i, and\ K_d$ are the proportional, integral, and derivative gains of the system. These gains are important and must be tuned manually to bring the system into control.

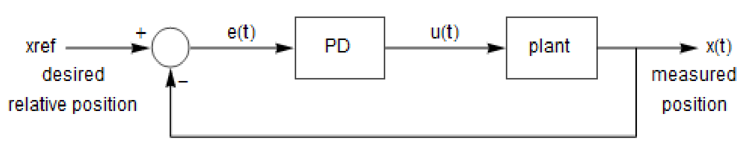

Figure 1.4: PID controller implementation

The PID controller acts on the plant, which, in this experiment, is the magnetic levitation system, and controls the position of the levitated magnet.

Methods

A fundamental part of this project is to develop the circuit that will be suitable for magnetic levitation. Each part is built around the arduino microcontroller that hosts the PID algorithm. The Hall effect sensor outputs a hall voltage that is sent to the analog pin of the arduino. The built in analog to digital converter on the arduino returns a value between 0 and 1024 for the Hall effect sensor. This value tells the PID algorithm where the magnet is relative to the electromagnet(current position). The algorithm then outputs the information digitally to the rest of the circuit. The arduino is constructed to send information from the PID algorithm through a pulse width modulated digital signal (PWM) which controls the electromagnet in the response to the levitated magnet. The algorithm constantly calculates the error between the set point and the current location of the magnet. The PID algorithm allows the user to adjust the values for set point and all of the gain parameters through the Arduino serial monitor so that the values can be seen in real time. (See Appendix B for schematic).

The PWM signal passes through an N-Channel MOSFET. The transistor's function is to amplify the PWM signal so that it will provide enough current to control the electromagnet. A diode is placed across the electromagnet so that any EMF will not damage it as the current in the electromagnet changes. Below, are the results for this experiment.

Data

Table 1: Physical parameters

x	Magnet position	0.47 - 2.3 cm Stable between 0.95-1.4 cm
i	Current in coil	30-100 mA
V	Input Voltage	9 Volts
m	Magnet mass	4.7e-5 kg
C	Magnetic constant	3.6225e-5 Nm/A^2.....
Rs	Resistor	1000 *ohms*
L	Inductance	15 mH

System builds

Figure 1.6 and Figure 1.7 are the prototype builds of the magnetic levitation system.

Figure 1.6: Prototype of the magnetic levitation device

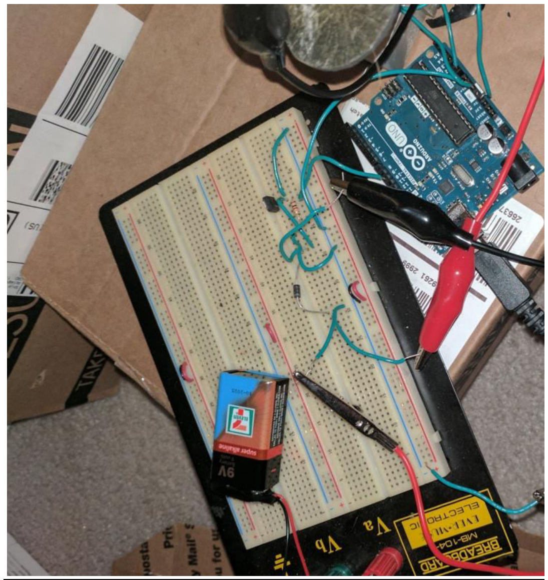

Figure 1.7: Arduino and breadboard with circuit

The final design of this system does not change any functionality of the original system and was rebuilt to create a more professional model.

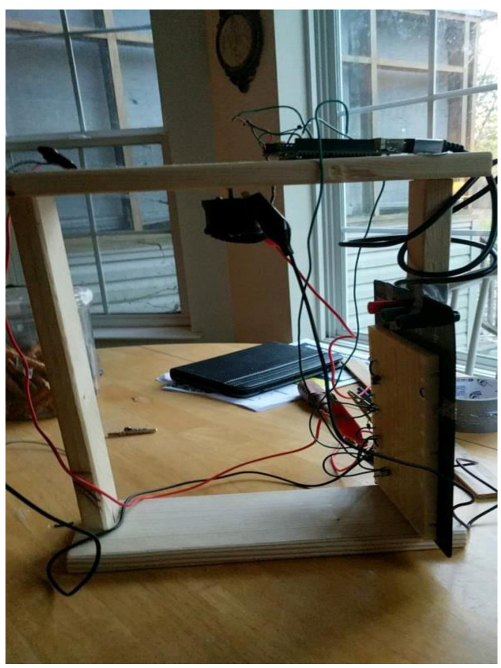

Figure 1.8: Final design of magnetic levitation system

Figure 1.9: Final magnetic levitation system in action

System response plots

This data is plotted using MATLAB. MATLAB can plot models of the system in state space form. The system is tuned using the PID_tuner tool in the MATLAB library.

Theoretical results

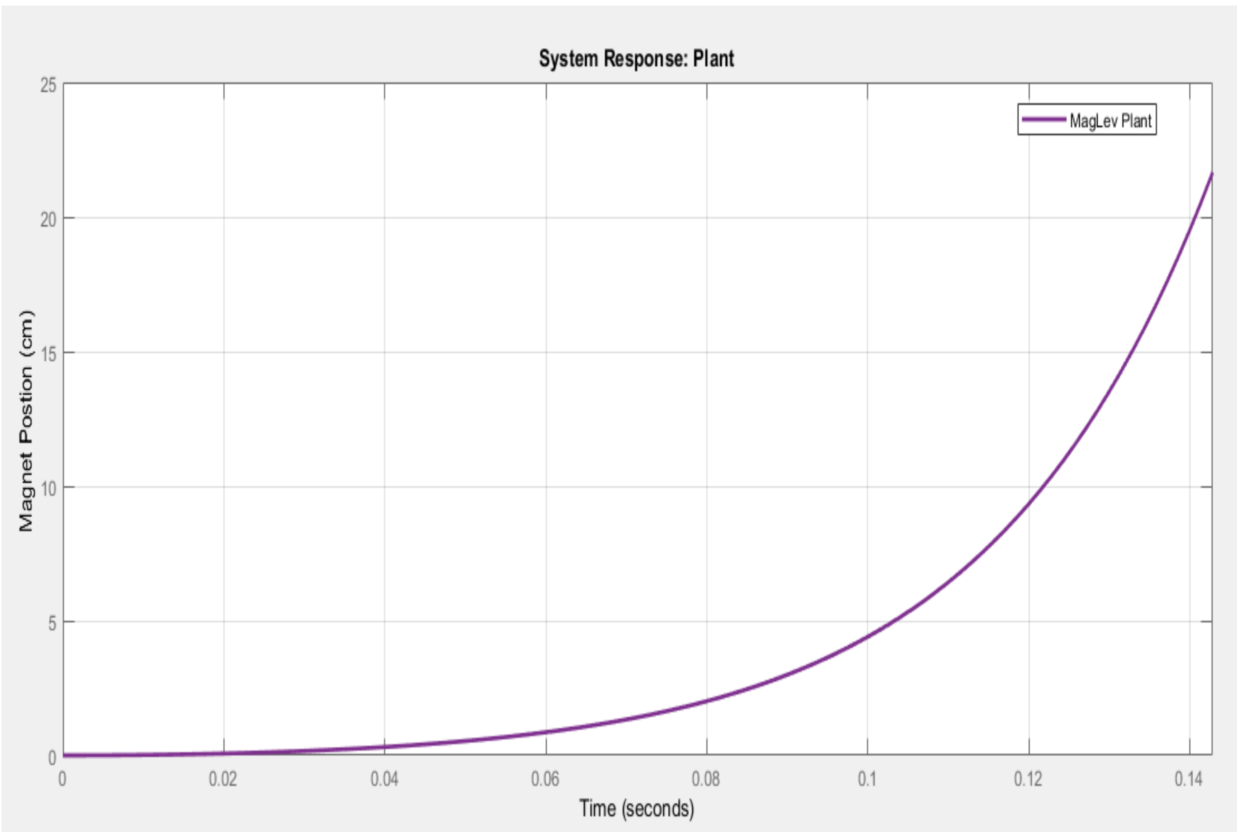

Figure 1.10: Plot of magnetic levitation system with not PID control

Figure 1.10 shows the effect of the magnetic levitation system without any PID control. The magnet will rise off toward infinity until it approaches the electromagnet because without the PID controller, there is no effective means of stabilization.

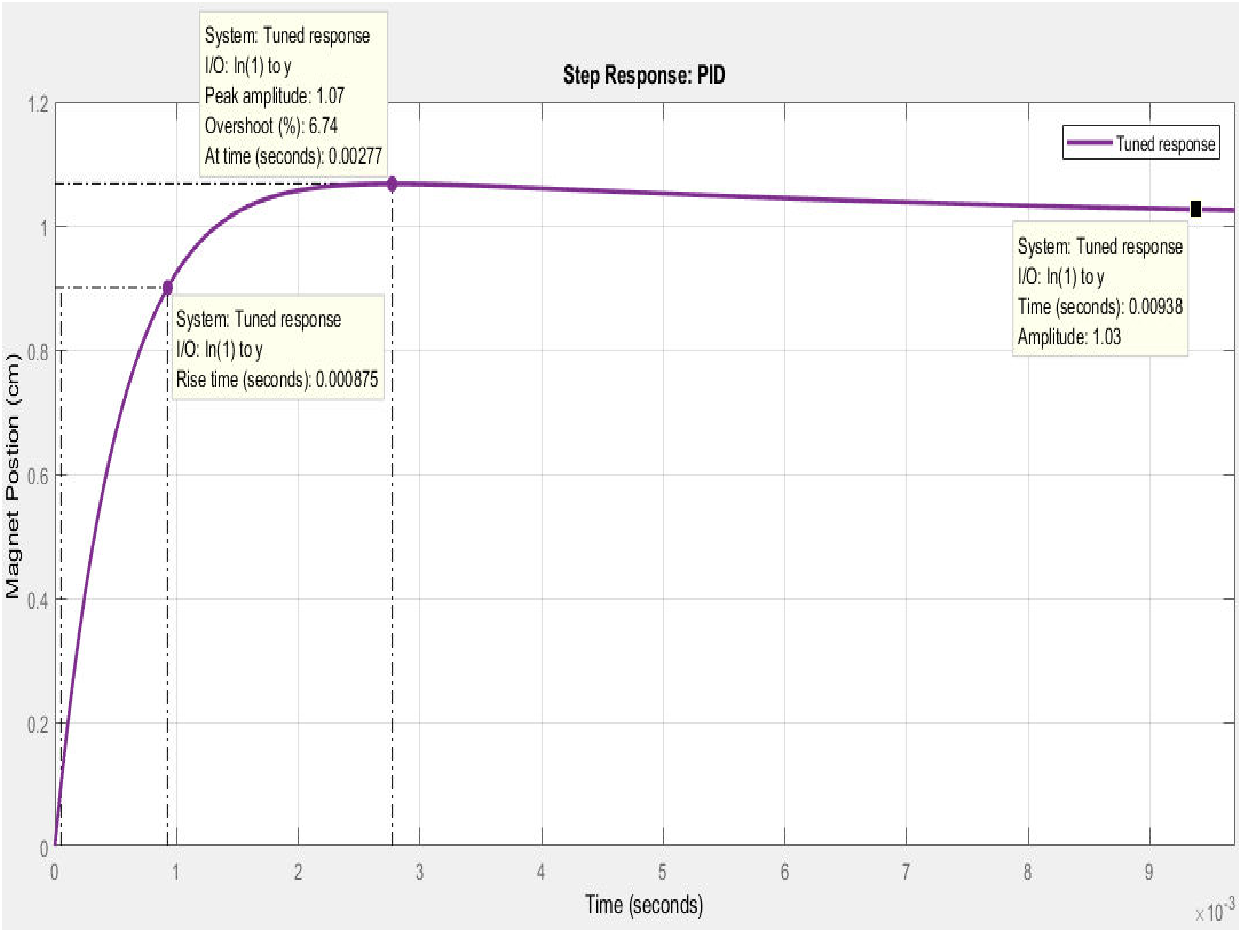

Figure 1.11: Plot of system with only proportional controller

Figure 1.11 shows the response of the system when the controller is implemented. The magnet attracts to the electromagnet, but is controlled to an equilibrium point at around 1.0 cm. This represents the best model given the PID tuned values (in Table 2).

<u>Actual results</u>

By tuning the PID values manually and plugging the values it into the model, the following figure is produced.

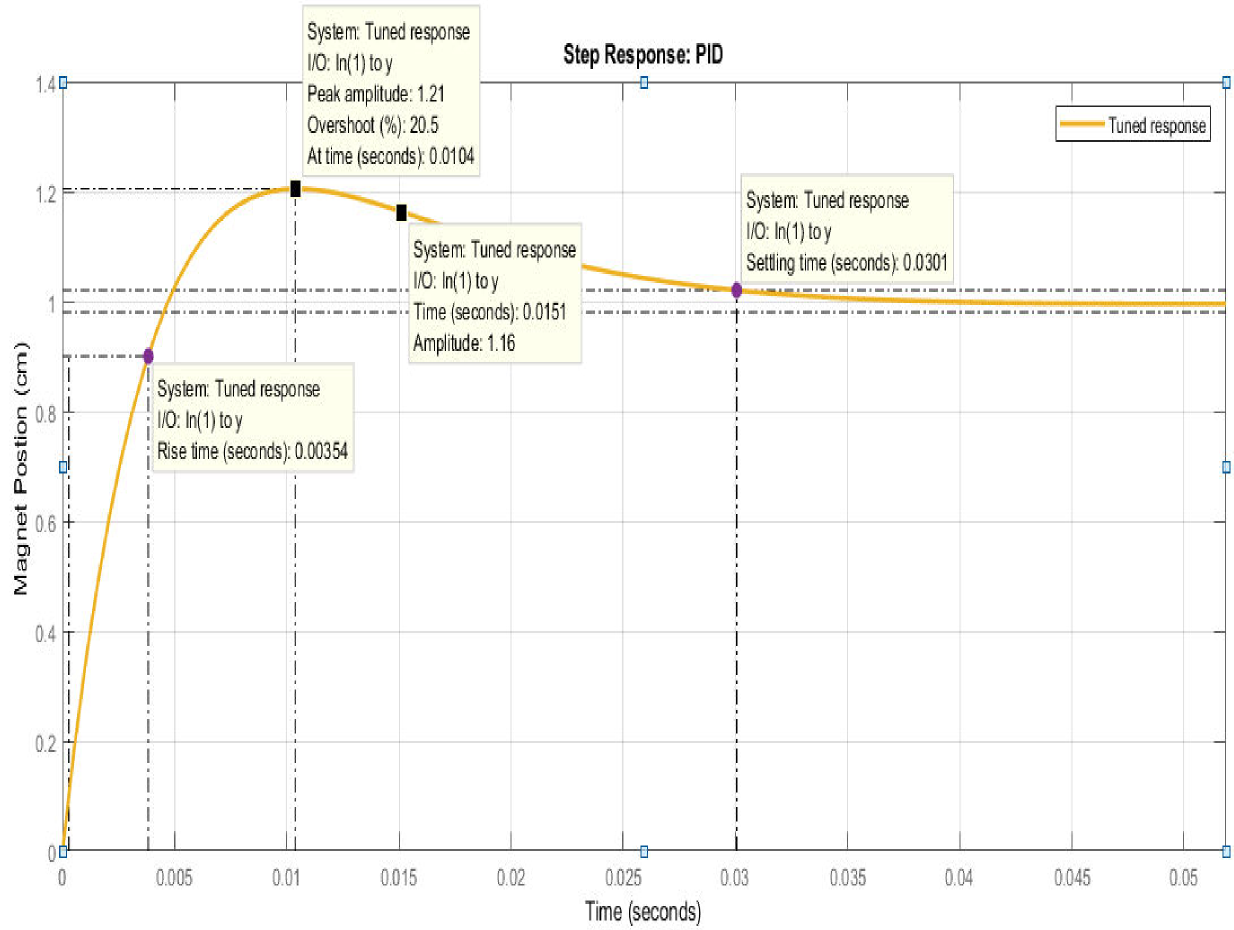

Figure 1.12 System response with PID tuning values manually

The theoretical and actual results are combined into one graph to show the effectiveness of the two models shown in figure 1.13

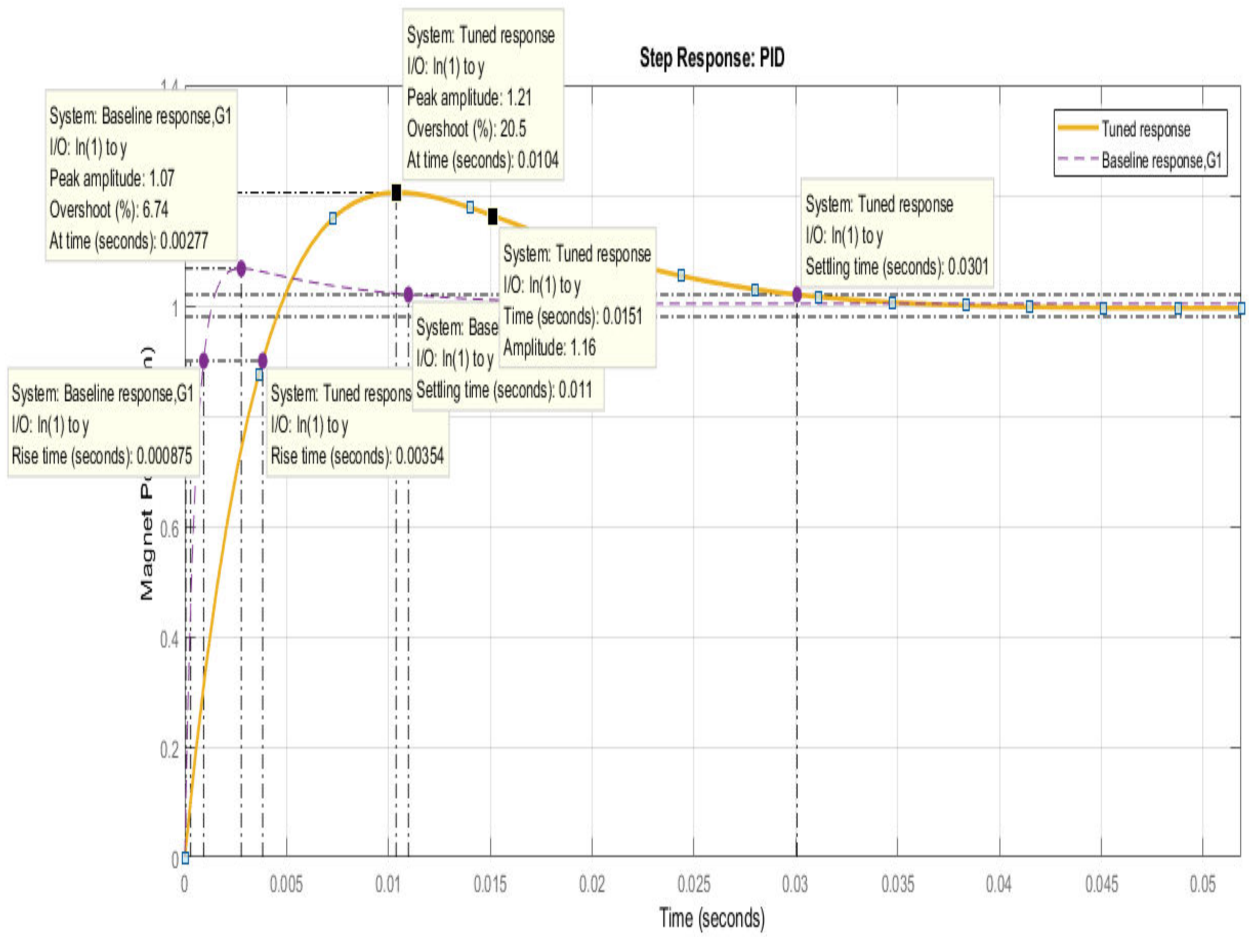

Figure 1.13 Theoretical and Actual results of PID acting on the system

Where the purple plot is the theoretical results and the yellow plot is the actual results. The results show that the actual model overshoots the theoretical model by 13.76 %. Function in MATLAB compute the values of the overshoot, settling time, and rising time

PID/ Model Values & Percent Error

Table 2: Theoretical & Actual values of system

	Theoretical	Actual	Percent Error
Peak Amplitude (cm)	1.07	1.21	13.1%
Settling Time (s)	0.011	0.0301	1.73%
Overshoot	6.74	20.5	45.1%

Table 3: PID gains

	Theoretical	Actual	Percent Error
K_p	26.3	36.7	39.5%
K_d	3.603	1.280	64.5%
K_i	78.77	57.49	27.0%

The largest amount of percent error comes from the PID gains. The reason for large amounts of percent error is due to the fact that the actual values were tuned manually while the theoretical gains were computed

Discussions & Conclusions:

The results of the experiment show that for correct implementation of PID control, that one can produce a system that levitates a magnet completely at a specific position. The magnet without a PID controller will get sucked up and cannot levitate (Figure 1.10). Adding a proportional gain, the system stars to oscillate, but still needs the aid of the derivative term and integral term. The system must use all three of the PID gains because the system will start to produce a steady state error and needs to be minimized to produce a steady state response at the equilibrium position. Figure 1.11 shows that the system stabilizes at 0.011 seconds when Kp = 23.6 , Kd= 3.603, and Ki = 78.77. This figure represents the best model for the system. The actual results for the system seen in figures 1.12 and 1.13 show that the theoretical model is consistent with the data. The actual model stabilizes at 0.0301 seconds, which is a bit slower than the original model, but is still very fast. The PID gains for the actual model are shown to be Kp = 36.7, Kd = 1.280, and Ki = 57.49. These gains are the best manually tuned values in this experiment, but still overshoot the theoretical model by 13.75 %. This amount of overshoot is not a problem because the magnet is still within stable range of the electromagnet. The system can be effectively match theoretical results when the gains are tuned to the values of the theoretical model. The system tends to work best when the magnet is placed between a distance of 0.95- 1.4 centimeters from the electromagnet. The PID will not work as the magnet gets too far or too close to the electromagnet because the strength of the magnetic field is weak when an object is too far or strong when an object is too close.

To further expand on this experiment, one could test the system using different masses that would in turn require different PID tuning values and responses. Another possible implementation of this project would be to build it using two hall effect sensors; one to detect the magnetic field at the top of the electromagnet and one at the bottom of the electromagnet to detect the magnetic field of both the electromagnet and the levitated magnet. Subtracting those two values would give a value that is directly proportional to the position. Using two hall effect sensors may produce a better accurate response due to the nature of the system and better tuning values. Some scholars have done this experiment using a Fuzzy Logic Controller(FLC) and comparing the results with a PID controller to see which controller works best. The magnetic levitation system is a highly unstable system, but with correct methodology, it can be done.

Appendix A: Materials & Resources

- UGN3503UA Linear Hall Effect Sensor
- N-Channel MOSFET Transistor (Model FQP30N06L)
- Arduino Uno (Rev 3)
- Renco RL-1256-4-15000 Electromagnet
- 1N4007 Diode
- A Breadboard
- 9 volt battery

Appendix B: Magnetic Levitation Schematic

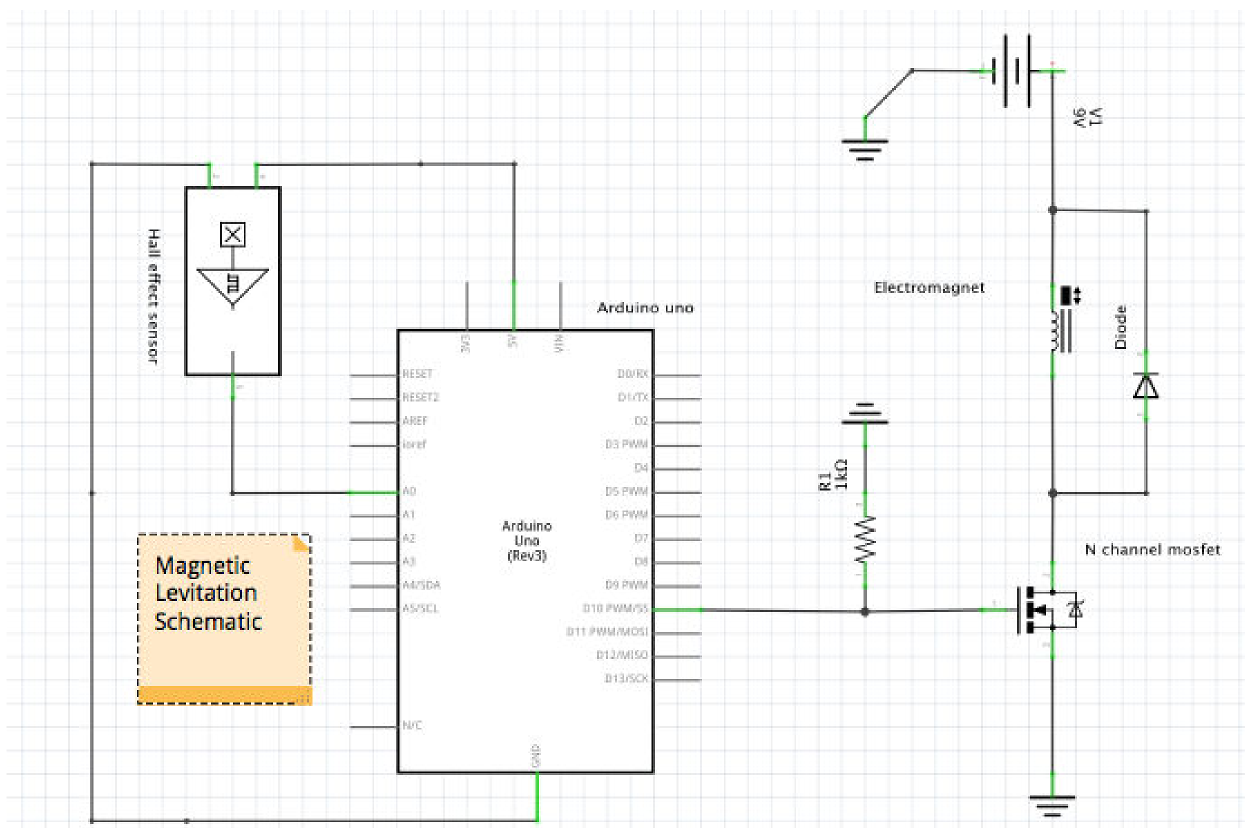

Figure 1.5: Magnetic Levitation Schematic

Bibliography:

- "Arduino - Getting Started." *Arduino* . N.p., n.d. Web. 12 Oct. 2016. <https://www.arduino.cc/en/Guide/HomePage>.
- CHOUDHARY , SANTOSH KR. . *Robust Feedback Control Analysis of Magnetic Levitation System*. PDF.
- Griffiths, David J. *Introduction to electrodynamics*. Noida, India: Pearson India Education Services, 2015. Print.
- Dr David Conner. Provided helpful code and information about PID.
- Dr. David Gore. Project Advisor
- "Introduction: PID Controller Design." *Control Tutorials for MATLAB and Simulink - Introduction: PID Controller Design*. N.p., n.d. Web. 10 Mar. 2017. <http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction§ion=ControlPID>
- "Linearized Control." *Linearized Control*. N.p., n.d. Web. 7 Mar. 2017. <http://web.mit.edu/2.737/www/MagLev/linearized/>.
- Nilsson, James William, and Susan A. Riedel. *Electric circuits*. Harlow, Essex: Pearson Education Limited, 2015. Print.
- "PID." *PID Theory - National Instruments*. N.p., n.d. Web. 20 Oct. 2016. <http://www.ni.com/white-paper/3782/en/>.
- "The Tuning of PID Loops." *Integrated Systems*. N.p., n.d. Web. 14 Jan. 2017. <https://innovativecontrols.com/blog/basics-tuning-pid-loops>.